

A Gesture-based CAPTCHA Design Supporting Mobile Devices

Nan Jiang
Bournemouth University
Fern Barrow

Poole, United Kingdom

+44 1202 962741

njiang@bournemouth.ac.uk

Huseyin Dogan
Bournemouth University
Fern Barrow

Poole, United Kingdom

+44 1202 962491

hdogan@bournemouth.ac.uk

ABSTRACT

In this paper we present the design and evaluation of a mobile user friendly CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) scheme based on our previous work. Unlike the commonly used character recognition based CAPTCHA schemes that require a user to type the distorted characters shown in an image to pass security checks, this scheme allows a user to use gestures to operate specific objects on the screen so as to complete a CAPTCHA quiz. Moreover, it uses partially processed or distorted textual instruction to prevent a bot from easily figuring out the objects to operate for retaining the security of the scheme as a bot cannot utilize context effects as easily as human users can. A comparative study is also conducted to understand the usability performance of this gestured based scheme against Google ReCAPTCHA, a popular character recognition scheme, where better usability has been reported with the former when both were used on smartphones.

Categories and Subject Descriptors

• Human-centered computing~Human computer interaction (HCI) • Human-centered computing~Interaction design • Human-centered computing~Ubiquitous and mobile devices • Human-centered computing~Gestural input • Human-centered computing~User interface design • Hardware~Touch screens • Security and privacy~Usability in security and privacy

Keywords

Gesture interaction; CAPTCHA; Context effect; Human Intelligence Proof (HIP); Mobile devices.

1. INTRODUCTION

Character recognition based CAPTCHAs are the most widely used security tests to tell computers and human apart on the web. However, such CAPTCHAs are often reported with poor usability as they are becoming more and more difficult for humans to solve as a trade-off of maintaining good level of security [4, 9]. This has also become a primary concern when these CAPTCHAs are used on mobile devices as they are not designed for accommodating the

hardware interface of common mobile devices [8, 15, 18].

In this paper we present a mobile user friendly CAPTCHA scheme that supports using touch interactions (gestures) to complete a CAPTCHA quiz based on our previous work [13]. Unlike character-recognition based CAPTCHAs that require a user to type the distorted characters shown in an image, this new CAPTCHA scheme asks a user to move specific objects around on the screen to solve a CAPTCHA quiz. At the same time, it utilizes context effects to retain the security of the scheme by partially and randomly processing the instruction of the quiz to prevent a bot from easily “guessing” out the required objects to operate. This is because a bot cannot utilize context effects to understand the meaning of the instruction as easily as human users can. We also conducted a comparative study between this gesture-based CAPTCHA and Google ReCAPTCHA and found better usability with the former.

The paper proceeds as follows. Section 2 presents the design concept and implementation process and Section 3 describes the experiment. Results are presented and discussed in Section 4 and 5 and conclusions are drawn in Section 6.

2. DESIGN

Early character recognition based CAPTCHAs like Gimpy took advantage of context effects [16] as humans are very good at visual perceptions when there presents a visual cue (context). However, with more and more advanced image processing and Artificial Intelligence (AI) algorithms being developed to break CAPTCHAs, complex visual recognition challenges have been introduced to minimize such security risks presented in this kind of CAPTCHAs. An example can be found in Figure 1 where the characters in the string, the length of the string, the size and orientation of each character are all randomized to reduce the probability to be solved by machines. This change has made character recognition tests become more and more difficult to human users as the context of establishing the appropriate recognition is now disappearing.

Enter the characters you see
New | Audio



Figure 1. Microsoft CAPTCHA.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.

British HCI 2015, July 13 - 17, 2015, Lincoln, United Kingdom

© 2015 ACM. ISBN 978-1-4503-3643-7/15/07...\$15.00

DOI: <http://dx.doi.org/10.1145/2783446.2783578>

The new CAPTCHA scheme keeps the original idea of utilizing context effects while making the challenge more mobile friendly by considering the main interactions methods offered by smartphones and other mobile devices with touchscreens.

2.1 Gesture-based Challenge

Existing approaches for designing mobile friendly CAPTCHAs suggest that touch interactions (gestures) should be considered as the foundation of a CAPTCHA scheme for mobile devices [16, 19]. This is because gesture controls such as tap, swipe, pinch and stretch are more intuitive on these devices [1, 17]. Therefore, the gesture based challenge in this CAPTCHA scheme is defined as manipulating specific objects on the screen according to randomly generated instructions. For example, a user needs to swipe a specific object to touch another specific object in order to complete a CAPTCHA quiz (Figure 2).

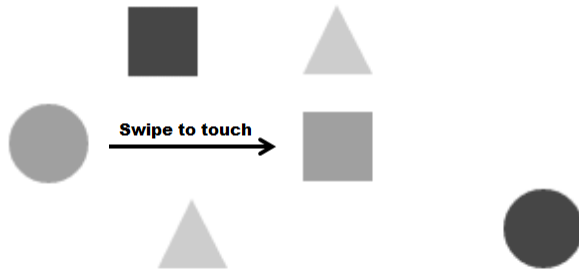


Figure 2. Gesture-based objects moving example.

2.2 Security Considerations

CAPTCHAs are used to tell humans and computers apart so the design goal of any effective CAPTCHA scheme is to make the CAPTCHA challenge easy to be solved by a human user but hard for a bot. Thus striking a balance between usability and security is always essential when designing a new CAPTCHA scheme. For an object manipulating challenge, the only way to retain security is to force a bot to take a brute force approach to try out all available objects one by one to solve the challenge instead of easily figuring out which specific objects should be moved. In this scheme, this is achieved by turning part of randomized instructions into distorted text (Figure 3).

Move the left object which is

The lightest pattern
to Touch the right object which is
Triangle

Figure 3. Processed instruction with distorted words.

First, this process will not affect human users from understanding the instruction as the context is still available. Evidence could be found in the user acceptance test for our novel design where all 17 participants showed positive responses to the partially processed instruction in terms of ease of understanding [13]. Second, since a bot cannot utilize context effects as easily as human users can, even when some distorted text has been recognized correctly, the bot will still have difficulties in understanding the actual meaning of the instruction as a whole so as to identify relevant objects to manipulate on the screen. With enough constraints being added to the success criteria of solving the challenge which are used to

form an instruction, it will eventually force the bot to take a brute force approach to move objects around to “guess” the success criteria and keep the mathematical probability to break the scheme extremely low. Note the mathematical probability to break such a scheme can be calculated by using the following formula:

$$P = \frac{1}{(T(T-1))^N}$$

Where P is the mathematical probability to break the CAPTCHA scheme when one object must be moved to touch another unmoved object in the scheme; T is the total number of objects and N is the number of tests presented in a CAPTCHA challenge. Taking FunCaptcha [11], an existing object moving CAPTCHA solution as an example (Figure 4) where eight objects are movable, the mathematical probability to solve this CAPTCHA without taking the instruction into consideration is $1/(8*(8-1)) = 1.79\%^1$. A practical consideration is that a CAPTCHA scheme is deemed to be broken when the attacker is able to reach a precision of at least 1% [5]. Here the precision is the fraction of CAPTCHAs answered correctly [3]. Considering the example given above, if two same type tests are used to form a complete CAPTCHA challenge, the probability will be dropped to $1.79\% * 1.79\% = 0.03\%$.

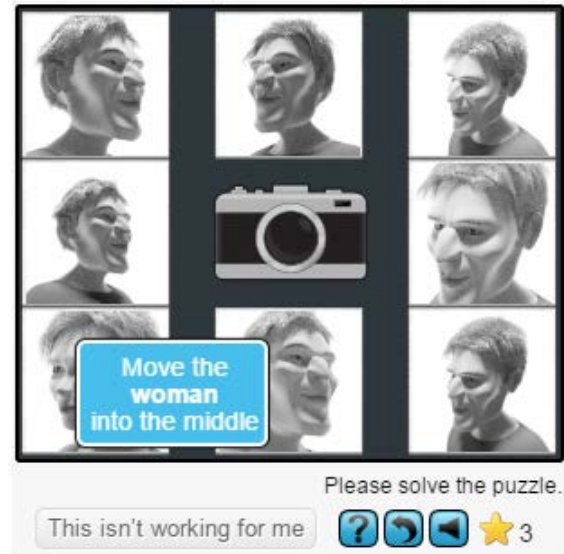


Figure 4. FunCaptcha demo.

In fact, less than 1% precision can be easily realized by either increasing the number of available objects on the screen for the challenge or introducing more sub tests in the challenge. For example, a challenge for moving one out of 11 available objects on the screen will bring the probability down to 0.91%. Moreover, including two small scale tests to form one gesture-based CAPTCHA challenge where each test is about moving one out of four available objects around will also reduce the probability to 0.694%.

1 FunCaptcha uses three different types of challenges: image recognition and moving challenge, image rotation challenge and image based cognition challenge. Figure 4 shows an image recognition and moving quiz where a user needs to move the woman into the middle (camera). In this example, there are eight movable objects as the central camera is not actually an object.

2.3 System Architecture and Implementation

Figure 5 shows the system architecture of creating such a gesture-based CAPTCHA scheme. Note that the success criteria are constraints to increase the difficulty for a bot to understand instruction properly and identify relevant objects like a human

user. For example, the instruction of “moving the object with the darkest shade on the left to the triangle with the lightest shade on the right” contains three explicit constraints: shade, shape and direction and one implicit constraint: number of available objects.

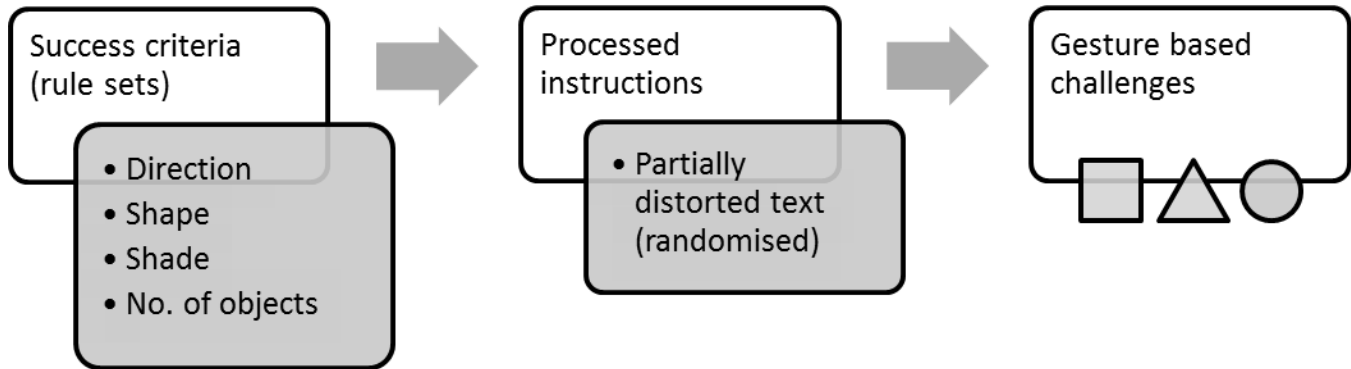


Figure 5. Gesture-based CAPTCHA system architecture.

This gesture-based CAPTCHA scheme can be easily implemented by using current frontend (e.g., JavaScript and HTML5 canvas) and backend technologies (e.g., PHP or Python). Frontend technologies are used to display the processed instruction and draw objects that can be controlled by gestures and backend technologies are used to define objects, set and render success criteria, and check gestures against these success criteria. The communication between frontend and backend can be implemented by using AJAX. An example implementation process is described below.

Step 1: Backend to generate a number of objects randomly based on predefined rules (e.g., shape, shade, number of objects etc.)

Step 2: Backend to send requirements to front-end to draw these objects with random coordinates on HTML5 canvas.

Step 3: Backend to decide the success criteria (e.g., moving a specific object to touch another specific object from the list of objects) and convert the criteria into instruction.

Step 4: Backend to distort part of instruction with any existing CAPTCHA library and output the processed instruction on the frontend.

Step 5: Frontend to communicate with backend when a user is moving an object to touch another by sending moved object ID and its coordinates to the backend when it collides with another object so that the backend can check if the movement meets the success criteria.

3. EXPERIMENT

A comparative study between the implementation of the new gesture based CAPTCHA scheme and an existing text-recognition based CAPTCHA solution was conducted. The purpose of this study was to understand the usability of the new CAPTCHA scheme on mobile phones.

3.1 Testing Environment

In order to simulate a real world validation process, two identical user login forms were created with gesture based CAPTCHA and a character recognition based CAPTCHA embedded respectively.

The gesture CAPTCHA scheme (Figure 6) supported 3 rule sets (i.e., shape, shade and the number of objects) with 3 objects on each side. A user will be asked to move an object from the left panel to touch another object on the right panel based on the random instructions where part of the text has been distorted.

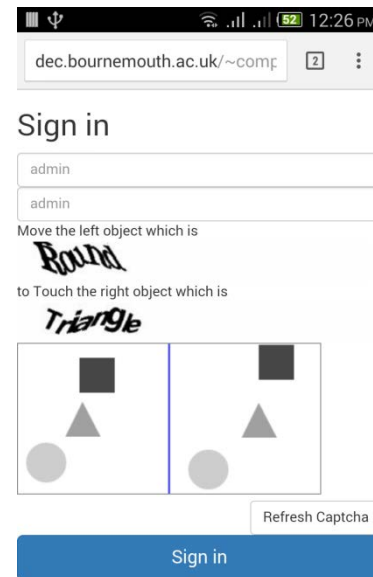


Figure 6. User login form with gesture-based CAPTCHA embedded.

Google ReCAPTCHA [12] was used as the character recognition CAPTCHA scheme for its popularity and flexibility of integration. The API considered in this implementation is the version 1 API (Figure 7). Note Google does not allow the customization of the actual recognition challenge being displayed on a website through its API so the default challenge (recognizing 5 characters) from a fresh setup was used during the comparison. Sometimes Google will adjust the difficulty of the recognition challenges based on a number of internal quality metrics to better safe-guard a website that uses its ReCAPTCHA service. When

this was happened in the experiment, for example, the challenge was changed to recognize two words or a door number plate, server refresh was always performed to make sure the challenges would return to the default setting.

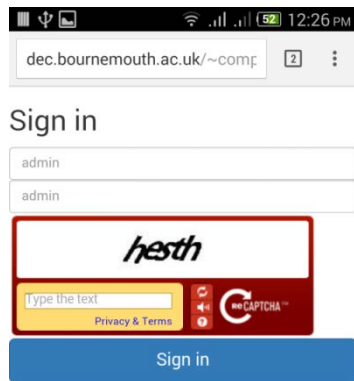


Figure 7. User login form with Google ReCAPTCHA embedded.

Since the experiment must be conducted with mobile devices, the smartphone used in this experiment is ZTE V9180, which is an Android empowered smartphone (Android kernel version: 4.4.2 KITKAT) equipped with a 5.0 inches capacitive touchscreen supporting resolution up to 720* 1280 pixels.

3.2 Measurement

Past research on understanding the usability issues of CAPTCHAs suggested that the accuracy, response time and perceived difficulty/satisfaction of using a CAPTCHA scheme [21] should be prioritized when running usability tests on CAPTCHAs. Therefore, four metrics include success rate, task time, errors and SUS (System Usability Scale) scores were used to measure the usability of both CAPTCHAs in this experiment.

3.3 Procedure

Each participant was explained about the purpose of the experiment and demonstrated how to use the two CAPTCHAs. They were also asked to try these CAPTCHAs a few times to familiarize themselves with the different CAPTCHA tests and the operation of the testing device so as to reduce the risks of having affected performance due to touch screen sensitivity or screen size. In addition to that, the odd number of participants were asked to perform 5 gesture CAPTCHA tests followed by 5 ReCAPTCHA tests and the even number of participants were asked to do 5 ReCAPTCHA tests and 5 gesture CAPTCHA tests for counterbalancing the results. The touchscreen of the testing smartphone was wiped clean in between two different sets of tests each participant needed to complete and after each participant completed all tests. The login credentials have been already hardcoded in the login form so participants only needed to focus on completing the CAPTCHA validations. Each participant's task performance was video recorded. After completion two sets of tasks, participants were asked to complete two SUS questionnaires, one for each CAPTCHA scheme and provide oral feedback about the tests.

4. RESULTS AND DISCUSSION

4.1 Participants

CAPTCHAs are widely used for validating user forms. This means the audience of CAPTCHAs should be generic web users rather than specific web users. Therefore, 20 participants from

undergraduate, postgraduate and academic staff between the ages of 18 and 48 were recruited to reflect the diversity of the user group. These participants completed a total of 100 gesture CAPTCHA tests and 100 ReCAPTCHA tests. All participants said they were familiar with text-recognition CAPTCHAs and owned a smartphone for daily use.

4.2 Success Rate (Completion Rate)

Success rate was measured as the percentage of CAPTCHA tests that users completed successfully. Since the outcome of a CAPTCHA validation can be either pass or fail, a similar binary measure was used to calculate the success rate. Figure 8 shows the success rate of gesture CAPTCHA and ReCAPTCHA completed by all participants where no participants failed any gesture CAPTCHA tests while 7 participants failed a total of 9 ReCAPTCHA tests (100% vs. 91%, $p < 0.05$). The high failure rate (9%) indicates that recognition difficulties found with character recognition schemes are still present and difficult to be resolved completely.

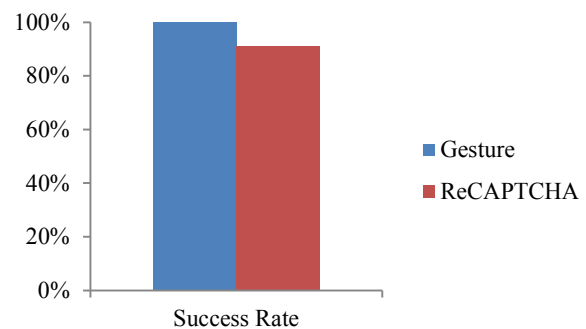


Figure 8. Success rate comparison.

4.3 Task Time

The task time was calculated by measuring how long it took a participant to complete a CAPTCHA test in seconds. The calculation started when a CAPTCHA test was properly displayed in a login page and finished as soon as a participant clicked the "sign in" button in the form. Table 1 shows the comparison between gesture CAPTCHA and ReCAPTCHA where participants performed significantly faster in gesture based CAPTCHA tests than ReCAPTCHA tests (t-test, $p < 0.00001$).

Table 1. Task time comparison

CAPTCHA	Task time (seconds)			
	Avg.	Min	Max	STD
Gesture	4.89	3	8	1.25
ReCAPTCHA	7.93	4	18	2.61

4.4 Errors

Errors in this experiment were defined as excessive touch interactions a participant made by comparing to the minimal number of touch interactions the participant would need for completing a test successfully. For gesture CAPTCHA, the minimal number of steps is 1 as user only needs to perform a swipe (from one object to another object) based on the instruction. For Google ReCAPTCHA, the optimal number of steps is 6 as user needs to tap on the CAPTCHA input field, type in five letters as shown in the CAPTCHA panel. Table 2 shows the errors and

type of errors made by users where gesture CAPTCHA reported significantly less errors than ReCAPTCHA. It is interesting to see that (1) all errors with gesture CAPTCHA scheme were related to object responsiveness and (2) most errors with ReCAPTCHA scheme were related to wrong operations. This indicates that the size of CAPTCHA challenges is important to the touch accuracy as suggested in [6, 20]. In addition, due to the lack of instruction on ReCAPTCHA, two participants did not realize the recognition challenge was case-insensitive so they tried to type a character exactly as it was shown the screen with capitals, which led to redundant operations.

Table 2. Error comparison

CAPTCHA	Errors	
	Total No.	Type of Errors
Gesture	6	Object responsiveness (no touch feedback from an object)
ReCAPTCHA	34	Wrong operations (choosing an irrelevant option accidentally)
	5	Wrong letter recognized (choosing wrong letter in the soft keyboard)
	8	Unnecessary operations (e.g., using shift for capitals; closing keyboard then clicking the login button)

4.5 SUS Score

SUS (System Usability Scale) is a widely used questionnaire to understand subjective assessment of a system [2]. In this experiment, participants were asked to complete the standard SUS questionnaire after the tests. Note in the standard SUS questionnaire where 10 questions are asked, question 1, 3, 5, 7 and 9 are favorable statements about the system while 2, 4, 6, 8, 10 are negative statements about the system. As a result, Gesture CAPTCHA scheme scored an overall score of 82 (min: 47.5, max: 100, STD: 12.48) and ReCAPTCHA scored 60.25 (min: 35, max: 100, STD: 19.85) where the detailed breakdowns are shown in Figure 9. It is clear that gesture CAPTCHA had better scores in 9 out of 10 questions. ReCAPTCHA achieved better score in Q4, which is one of the two learnability related questions [14].

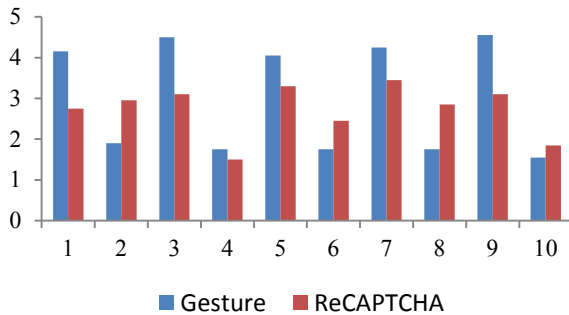


Figure 9. SUS score breakdowns.

5. IMPLICATIONS

ReCAPTCHA is a representative character recognition based CAPTCHA scheme so the tests on this scheme can be used to

provide guidance on understanding the overall usability performance of this gesture-based CAPTCHA scheme against character recognition based CAPTCHA schemes. Experimental results show that the new gesture CAPTCHA scheme is generally more mobile user friendly than character recognition CAPTCHA scheme in terms of efficiency, effectiveness and user satisfaction. This is because gesture-based CAPTCHA schemes, which rely on touch interactions, are more natural than character recognition based schemes when used on mobile devices. However, it should be noted that the minimum task time for completing gesture and ReCAPTCHA challenges was not significantly different (3 seconds vs. 4 seconds). The post-video analysis noted that the efficiency of solving character recognition challenges was strongly related to the participants' typing habits. That is, those using two hands on the soft keyboard to solve ReCAPTCHA challenges performed generally faster than those with one hand. This finding is supported by participants' feedback after the tests where four participants specifically pointed that they were very used to texting or messaging other people through Instant Messaging (IM) apps on smartphones so they could type very fast. In addition, although participants performed significantly faster in the gesture-based CAPTCHA tests than ReCAPTCHA tests, it should note the following facts:

- 1 The test performance in the former was measured based on manipulating one out of six objects where the probability for breaking the CAPTCHA scheme was 3.33%. This is higher than the recommended precision for breaking a CAPTCHA scheme which is 1%.
- 2 The performance in the latter was measured based on recognizing five characters so results may vary when the complexity of the challenges changes.

Therefore, more tests on different settings will be needed to further understand how to design usable and secure gesture-based CAPTCHA schemes.

Last but not least, unlike ReCAPTCHA which did not provide any instruction on how to use it, user would need to understand the instruction provided in the gesture based CAPTCHA before solving the challenges. A pause to touch the screen before a participant started to solve the gesture CAPTCHA scheme has been noted in the video analysis. This pause was related to the time the participant needed to read and understand the processed instruction. It was found that the length of pause was mainly dependent on (1) the length of the text that was distorted and (2) if the participant is a native English speaker. This suggests that a localized gesture based CAPTCHA scheme may be considered if the target audience is not native English speakers [10].

6. CONCLUSION

In this paper, we presented the design of a mobile user friendly CAPTCHA scheme that supports using touch interactions (gestures) to solve a CAPTCHA challenge. The comparative study has reported some preliminary but promising results with this gesture based CAPTCHA scheme. However, we also identified some design issues (e.g., touch responsiveness) which need to be improved in the future. Moreover, the comparative study did not consider any existing touch or gesture based CAPTCHA schemes such as FunCaptcha, Picatcha and KeyCaptcha. A future study comparing this gesture-based CAPTCHA scheme with existing solutions needs to be done to further investigate the usability of the scheme. The CAPTCHA implementation used in the experiment only considered limited

accessibility requirements (i.e., greyscale shades instead of colored shades), more improvements on accessibility also needs to be considered to consolidate this design.

7. ACKNOWLEDGEMENTS

The authors would like to thank all students and colleagues who volunteered to take part in the comparative study. They would also like to thank BCS 2013 Work-in-Progress reviewers and BCS 2014 Full paper reviewers for providing invaluable feedback to consolidate this work.

8. REFERENCES

- [1] Android. Gestures | Android Developers. 2015. Retrieved March 2, 2015 from <http://developer.android.com/design/patterns/gestures.html>.
- [2] Brooke, J. 1996. SUS - A quick and dirty usability scale. In *Usability Evaluation in Industry*, Jordan, P.W., Thomas, B., Weerdmeester, B.A. and McClelland, A.L. (eds.). Taylor & Francis, London, UK, 189 - 194.
- [3] Bursztein, E. and Bethard, S. 2009. DeCAPTCHA: breaking 75% of eBay audio CAPTCHAs. In *Proceedings of the 3rd USENIX conference on Offensive technologies* (Montreal, Canada, August 10 - 14, 2009). WOOT '09. USENIX Association, Berkeley, CA, USA, 8-8.
- [4] Bursztein, E., Bethard, S., Fabry, C., Mitchell, J.C. and Jurafsky, D. 2010. How good are humans at solving CAPTCHAs? A large scale evaluation. In *Proceedings of the 31st IEEE Symposium on Security and Privacy* (Oakland, CA, USA, May 16 - 19, 2010). SP '10. IEEE Computer Society, Washington, DC, USA, 399-413. DOI=<http://dx.doi.org/10.1109/SP.2010.31>.
- [5] Bursztein, E., Martin, M. and Mitchell, J. 2011. Text-based CAPTCHA strengths and weaknesses. In *Proceedings of the 18th ACM Conference on Computer and Communications Security* (Chicago, IL, USA, October 17 - 21, 2011). CCS '11. ACM, New York, NY, 125-138. DOI=<http://doi.acm.org/10.1145/2046707.2046724>.
- [6] Bursztein, E., Moscicki, A., Fabry, C., Bethard, S., Mitchell, J. C. and Jurafsky, D. 2014. Easy does it: more usable CAPTCHAs. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Canada, April 26 - May 1, 2014). CHI '14. ACM, New York, NY, 2637 - 2646. DOI=<http://doi.acm.org/10.1145/2556288.2557322>.
- [7] Chellapilla, K., Larson, K., Simard, P. Y. and Czerwinski, M. 2005. Building segmentation based human-friendly human interaction proofs (HIPs). In *2nd International Workshop on Human Interactive Proofs* (Bethlehem, PA, USA, May 19 - 20, 2005). HIP 2005. Springer, Berlin, Heidelberg, 1-26.
- [8] Chow, R., Golle, P., Jakobsson, M., Wang, L. and Wang, X. 2008. Making CAPTCHAs clickable. In *Proceedings of the 9th Workshop on Mobile Computing Systems and Applications* (Napa, CA, USA, February 25 - 26, 2008). HotMobile '08. ACM, New York, 91 - 94.
- [9] Fidas, C. A., Voyiatzis, A. G. and Avouris, N. M. 2011. On the necessity of user-friendly CAPTCHA. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vancouver, BC, Canada, May 7 - 12, 2011). CHI '11. ACM, New York, NY, 2623-2626. DOI=<http://doi.acm.org/10.1145/1978942.1979325>.
- [10] Fidas, C. and Voyiatzis, A. G. 2013. On Users' Preference on Localized vs. Latin-Based CAPTCHA Challenges. In *Proceedings of the 14th IFIP International Conference on Human-Computer Interaction* (Cape Town, South Africa, September 2 - 6, 2013). INTERACT 2013. Springer, Berlin Heidelberg, 358 - 365.
- [11] FunCaptcha. FunCaptcha – the best Captcha alternative. 2015. Retrieved March 2, 2015 from <http://www.funCAPTCHA.co/CAPTCHA/>.
- [12] Google. reCAPTCHA: Easy on Humans, Hard on Bots. 2015. Retrieved March 2, 2015 from <http://www.google.com/recaptcha/intro/index.html>.
- [13] Jiang, N. and Tian, F. 2013. A Novel Gesture-based CAPTCHA Design for Smart Devices. In *Proceedings of the 27th International BCS Human Computer Interaction Conference* (London, UK, September 9 - 13, 2013), HCI 2013. British Computer Society, London, 49.
- [14] Lewis, J. R. and Sauro, J. 2009. The factor structure of the system usability scale. In *Human Centered Design*, Springer Berlin Heidelberg, 94-103.
- [15] Lin, R., Huang, S. Y., Bell, G. B., and Lee, Y. K. 2011. A new CAPTCHA interface design for mobile devices. In *Proceedings of the 12th Australasian User Interface Conference* (Perth, Australia, January 17 - 20, 2011). AUIC 2011. Australian Computer Society, Sydney, Australia. 3 - 8.
- [16] McClelland, J. L. and Rumelhart, D. E. 1981. An interactive activation model of context effects in letter perception: I. An account of basic findings. *Psychological Review*, 88, 5 (Sept. 1981), 375.
- [17] Microsoft. Gestures, manipulations, and interactions. 2015. Retrieved March 2, 2015 from <https://msdn.microsoft.com/en-us/library/windows/apps/hh761498.aspx>.
- [18] Okada, M. and Matsuyama, S. 2012. New CAPTCHA for smartphones and tablet PC. In *Proceedings of the 9th IEEE Consumer Communications and Networking Conference* (Las Vegas, NV, USA, January 14 - 17, 2012). CCNC 2012. IEEE Computer Society, Washington, DC, USA, 34 - 35.
- [19] Reynaga, G. and Chiasson, S. 2013. The Usability of CAPTCHAs on Smartphones. In *Proceedings of 10th International Conference on Security and Cryptography* (Colmar, Alsace, France, July 20 - 22, 2013), SECRYPT 2013. SciTePress, 427 - 434.
- [20] Reynaga, G., Chiasson, S. and van Oorschot, P. C. 2015. Exploring the Usability of CAPTCHAs on Smartphones: Comparisons and Recommendations. In *Proceedings of 2015 Network and Distributed System Security (NDSS) Symposium* (San Diego, CA, USA, February 8 - 11, 2015). NDSS '15. Internet Society, Reston, VA, USA.
- [21] Yan, J. and El Ahmad, A. S. 2008. Usability of captchas or usability issues in captcha design. In *Proceedings of the 4th symposium on Usable privacy and security* (Pittsburgh, PA, USA, July 23 - 25, 2008). SOUPS '08. ACM, New York, NY, USA, 44-52. DOI=<http://doi.acm.org/10.1145/1408664.1408671>.